CS 280
Programming Language Concepts

Files

What Is A File?

- You probably think of a file as a collection of bytes stored on some permanent storage, like a disk
- Think about this more broadly. A file is an interesting and useful concept that is available to programmers in most languages and operating systems. It does not have to be on a disk.

Files as streams of bytes

- From the standpoint of your program, you really just want to be able to read input from somewhere and write output to somewhere
- Think of this as a stream of bytes that come into your program or go out of your program
- You "read" input from some stream of bytes, and you "write" output to some stream of bytes
- This abstraction of a stream of bytes is a file

What is a file, really?

- It might be an actual file on a disk
  - An actual disk drive
  - A memory stick (not really a disk)
  - An SSD drive (also not really a disk)
- It might be a stream of bytes from the network
- It might be a "special file" – this is a Unix concept for things like the console or the screen

What is "buffering"?

- Your program might only want to read or write a byte at a time
- That might not be the most efficient way to do input/output
- Sometimes it makes sense to read things in larger chunks, even though the user of the data is only using things a byte at a time
- This is what "buffering" means
  - Reading a disk probably reads a large block of data
  - Writing a disk writes a block at a time
  - Networking reads and writes packets, not a byte at a time

File I/O

- C++ allows you to create a stream that is associated with a file
- You must open a file before reading from it and close it when done.
- #include <fstream> for access to file streams

Reading from files

- An ifstream is an Input File stream
- Once you open the stream, you can read it the same way that you read from cin
  - in fact, cin is just a stream reading from the "standard input" file, not a file that you explicitly open
- You should close a file when you are finished

- You can write to a file with an ofstream (Output File Stream)
- You open an ofstream for writing, and you write to it the same way that you write to cout
- You should close the file when you are finished

Code sample

```
ifstream infile;
infile.open(argv[1]);

if( infile.is_open() == false ) {
      cerr << "Couldn't open!" << endl;
      return -1;
}

string word;
infile >> word; // read one word from the file

infile.close(); // done
```